WHAT IS CLAIMED IS:

1. A method of representing a structure for a message for a software object corresponding to an application program interface (API) within a graphical user interface, the method comprising:

retrieving a code for the message structure of the API, where the code includes nodes with node names and delimiters, where the delimiters indicate a hierarchical structure for the nodes;

determining a hierarchical structure of the code of the message structure for the API; and

displaying the message structure of the API in a pane of the graphical user interface in a hierarchical according to the hierarchical structure of the message.

2. The method as defined in Claim 1, wherein a node corresponds to an API function.

3. The method as defined in Claim 1, wherein a node from the message is selected from the group consisting of element nodes, field nodes, method nodes, and parameter nodes, and where an element node can have a sub-node selected from the group consisting of an element, field, method, and parameter, a field node has no sub-nodes, a method node can have a sub-node of parameter type, and a parameter node has no sub-nodes.

4. The method as defined in Claim 1, wherein the code is a text-based code, where determining the hierarchical structure further comprises parsing the text-based code according to predefined delimiters embedded in the text-based code.

5. The method as defined in Claim 1, further comprising:

determining when a pointer that is manipulated by a pointing device is guided over a node symbol and when an input from the pointing device indicates a selection by the pointing device; and

expanding the node corresponding to the selected node symbol to display sub-nodes of the node in the pane.

6. The method as defined in Claim 5, wherein the input from the pointing device corresponds to a mouse button click.

7. The method as defined in Claim 1, further comprising:

determining when a pointer that is manipulated by a pointing device is guided over a node symbol and when an input from the pointing device indicates a selection by the pointing device; and

collapsing the node corresponding to the selected node symbol such that sub-nodes of the selected node are not displayed in the pane.

8. The method as defined in Claim 1, further comprising:

detecting that a pointer that is manipulated by a pointing device is guided over a portion of a displayed node in the hierarchical tree and that an input from the pointing device has been received such that the displayed node is selected; and

displaying a list of properties for the selected node of a displayed hierarchical tree in a separate pane of the graphical user interface in response to the selection.

9. A computer-readable medium having computer-executable instructions for performing a method of representing a structure for a message for a software object corresponding to an application program interface (API) within a graphical user interface, comprising:

retrieving a code for the message structure of the API, where the code includes node names and delimiters, where the delimiters indicate a hierarchical structure;

determining a hierarchical structure of the code of the message structure for the API; and

displaying the message structure of the API in a pane of the graphical user interface in a hierarchical according to the hierarchical structure of the message.

10. A system that represents a structure of a message for a software object corresponding to an application program interface (API), the system comprising:

a graphical user interface;

a module configured to retrieve a code for the message structure of the API, where the code includes nodes with node names and delimiters, where the delimiters indicate a hierarchical structure for the nodes;

a module configured to determine a hierarchical structure of the code of the message structure for the API; and a module configured to display the message of the software object in a pane of the graphical user interface in a hierarchy according to the hierarchical structure of the message.

11. The system as defined in Claim 10, further comprising:

a module configured to determine when a pointer that is manipulated by a pointing device is guided over a node symbol and when an input from the pointing device indicates a selection by the pointing device; and

expanding the node corresponding to the selected node symbol to display sub-nodes of the node in the pane.

12. The system as defined in Claim 10, wherein the input from the pointing device corresponds to a mouse button click.

13. A method of generating a structure for a message for making an application program interface (API) call in a graphical user interface, the method comprising:

displaying a representation for a root node for the message structure in response to a request to display a new message structure, where the representation appears in a pane of the graphical user interface;

receiving a request to add a child node to the root node of the message structure, where the child node is selected from the group consisting of element nodes, field nodes, method nodes, and parameter nodes;

displaying a representation of the added child node to the root node such that a hierarchical relationship between the added child node and the root node is illustrated in a hierarchical tree; and

automatically generating code for the message structure, wherein the code includes embedded delimiters that indicate the hierarchical structure of corresponding messages for the API calls.

14. The method as defined in Claim 13, further comprising:

receiving a request to add a sub-node to the child node; and

displaying a representation of the added sub-node such that the hierarchical relationship between the added sub-node and the child node is illustrated in a hierarchical tree.

15. The method as defined in Claim 13, further comprising revising a message structure, the method comprising:

detecting that a pointer that is manipulated by a pointing device is guided over a portion of a displayed node in the hierarchical tree;

receiving an indication that a button on the pointing device has been selected and continues to be selected when the pointer is over the portion of the displayed node;

detecting movement of the pointer by the pointing device;

visually dragging the displayed node in correspondence with the movement of the pointer;

receiving an indication that the button on the pointing device has been released;

detecting that the pointer has moved from an original point in the hierarchical tree to a second point in the hierarchical tree, where the second point corresponds to a location of the pointer when the button had been released;

visually dragging the displayed node from the original point to the second point; and

automatically regenerating code for the message structure, wherein the code includes embedded delimiters that indicate the hierarchical structure of the revised message structure.

16. The method as defined in Claim 13, further comprising:

detecting that a pointer over a portion of a displayed node in the hierarchical tree and that an input from a pointing device has been received such that the displayed node is selected; and

displaying a list of properties for the selected node in a separate pane of the graphical user interface in response to the selection of the node.

17. The method as defined in Claim 13, wherein the child node that is added is selected from a pre-programmed API function.

18. The method as defined in Claim 13, wherein the child node that is added is selected from a pre-programmed interface method.

19. A computer-readable medium having computer-executable instructions for performing a method of generating a structure for a message for making an application program interface (API) call comprising:

displaying a representation for a root node for the message structure for the API call in response to a request for a new message structure, where the representation appears in a pane of the graphical user interface;

receiving a request to add a child node to the root node of the message structure, where the child node is selected from the group consisting of element nodes, field nodes, method nodes, and parameter nodes;

displaying a representation of the added child node to the root node such that a hierarchical relationship between the added child node and the root node is illustrated in a hierarchical tree; and

automatically generating code for the message structure, wherein the code includes embedded delimiters that indicate the hierarchical structure of the message structure.

20. A system of generating a structured message for making an application program interface (API) call, the system comprising:

a graphical user interface;

a module configured to display a representation for a root node for the message structure in response to a request for a new message structure, where the representation appears in a pane of the graphical user interface;

a module configured to receive a request to add a child node to the root node of the message structure, where the child node is selected from the group consisting of element nodes, field nodes, method nodes, and parameter nodes;

a module configured to display a representation of the added child node to the root node such that a hierarchical relationship between the added child node and the root node is illustrated in a hierarchical tree; and

a module configured to automatically generate code for the message structure, wherein the code includes embedded delimiters that indicate the hierarchical structure of corresponding messages for the API calls.

21. The system as defined in Claim 20, further comprising:

a module configured to detect when a pointer that is manipulated by a pointing device is guided over a portion of a displayed node in the hierarchical tree;

a module configured to detect that a button on the pointing device has been selected and continues to be selected when the pointer is over the portion of the displayed node;

a module configured to detect a position of the pointer;

a module configured to visually drag the displayed node in the graphical user interface in correspondence with motion of the pointer;

a module configured to receive an indication that the button on the pointing device has been released;

a module configured to detect that the pointer has moved from an original point in the hierarchical tree to a second point in the hierarchical tree, where the second point corresponds to a location of the pointer when the button had been released;

a module configured to visually drag the displayed node from the original point to the second point; and

a module configured to automatically regenerate code for the structured message, wherein the code includes embedded delimiters that indicate the hierarchical structure of the structured message such that the recalculated code reflects the change in position for the node from the original point to the second point.

22. A method of automatically generating code for translating structured messages for API calls with a graphical user interface, the method comprising:

displaying a first hierarchical tree to represent a source event type definition in the graphical user interface;

displaying a second hierarchical tree to represent a destination event type definition in the graphical user interface;

receiving an indication of a selection from a pointing device;

detecting that a pointer in the graphical user interface that is manipulated by the pointing device is over a representation of a first node in the first hierarchical tree at the time that selection from the pointing device was initiated;

detecting that the pointer is guided from the representation of the first node in the first hierarchical tree to a representation of a second node in the second hierarchical tree, where the representation of the second node corresponds to a location of the pointer in the graphical user interface at the time that selection from the pointing device was terminated;

drawing a line from the representation for the first node in the first hierarchical tree to the representation of the second node in the second hierarchical tree; and

automatically generating software code that maps data from the first node of the source event type definition to the second node of the destination event type definition.

23. The method as defined in Claim 22, wherein the pointing device corresponds to a mouse device, and wherein receiving the indication of the selection from the pointing device further comprises detecting a click of a mouse button.

24. The method as defined in Claim 22, further comprising displaying software code in a pane of the graphical user interface, where the displayed software code includes the automatically generated software code.

25. The method as defined in Claim 24, further comprising:

receiving a second indication of a selection from the pointing device;

detecting that the pointer is over a displayed line of code at the time of the second indication;

retrieving rule properties relating to the selected line of code; and

displaying the retrieved rules in the graphical user interface.

26. The method as defined in Claim 22, wherein drawing the line from the representation for the first node in the first hierarchical tree to the representation of the second node in the second hierarchical tree further comprises:

drawing the line in the graphical user interface while the indication of the selection is present, where a first end of the line starts from about the first representation for the first node in the first hierarchy, and where a second end of the line ends about at the pointer such that the second end of the line moves with movement of the pointer; and

fixing the line at the second node of the second hierarchical tree when the indication of the selection is terminated.

27. The method as defined in Claim 26, wherein the pointing device corresponds to a mouse device, where the indication that the selection is present corresponds to a detection that a user is holding down a mouse button on the mouse device, and where the indication that the selection is terminated corresponds to a detection that a user has released the mouse button on the mouse device.

28. The method as defined in Claim 22, wherein the source event type definition and the destination event type definition both correspond to structured messages, and where the displayed representations of the source event and the destination event and the displayed lines interconnecting representations of nodes from the source event type definition to the destination event type definition correspond to a translation of the structured message of the source event type definition to the structured message of the destination event type definition from one application with an API to another application with another API.

29. The method as defined in Claim 22, wherein the source event type definition corresponds to an API function and the destination event type definition corresponds to an interface method with stored procedures for a database, where the generated code is adapted to retrieve information from the database via a stored procedure for the API function.

30. The method as defined in Claim 22, wherein the source event type definition corresponds to an object and the destination event type definition corresponds to an interface method for a database, where a representation of a parameter of a method of the object is graphically connected in the graphical user interface to a representation of a stored procedure

of the interface method, and where the generated code is adapted to retrieve a value for the parameter from the database via the stored procedure.

31. The method as defined in Claim 22, wherein the graphical user interface is organized such that the source event type definition and the destination event type definition are displayed in separate panes of the graphical user interface.

32. The method as defined in Claim 22, wherein the graphical user interface is organized such that the source event type definition is displayed in a first pane, the destination event type definition is displayed in a second pane, and the first pane is to the left of the second pane.

33. The method as defined in Claim 22, wherein the graphical user interface comprises a first pane, a second pane, and a third pane, where the first pane displays a representation of the source event type definition, where the second pane displays a representation of the destination event type definition, where the third pane is disposed between the first pane and the second pane, and where the third pane displays a correspondence between a node of the source pane and a node of the destination pane via a line drawn between the first pane and the second pane.

34. The method as defined in Claim 22, wherein the graphical user interface further comprises a mapping pane, which is disposed between a source event pane and a destination event pane, where a portion of a line in the mapping pane joins a portion of a line from the source event pane and a portion of a line from the destination event pane to illustrate a correspondence of data between a source event node and a destination event node.

35. The method as defined in Claim 22, further comprising:

providing a pane in the graphical user interface configured to show the automatically generated software code that maps data from the first node of the source event type definition to the second node of the destination event type definition; and

permitting the automatically generated source code to be selected and modified for customizations.

36. A computer-readable medium having computer-executable instructions for performing a method of automatically generating code for translating structured messages for API calls comprising:

displaying a first hierarchical tree to represent a source event type definition in the graphical user interface;

displaying a second hierarchical tree to represent a destination event type definition in the graphical user interface;

receiving an indication of a selection from a pointing device;

detecting that a pointer in the graphical user interface that is manipulated by the pointing device is over a representation of a first node in the first hierarchical tree at the time that selection from the pointing device was initiated;

detecting that the pointer is guided from the representation of the first node in the first hierarchical tree to a representation of a second node in the second hierarchical tree, where the representation of the second node corresponds to a location of the pointer in the graphical user interface at the time that selection from the pointing device was terminated;

drawing a line from the representation for the first node in the first hierarchical tree to the representation of the second node in the second hierarchical tree; and

automatically generating software code that maps data from the first node of the source event type definition to the second node of the destination event type definition.

37. A system that automatically generates code for translating structured messages for API calls comprising:

a graphical user interface;

a module configured to display a first hierarchical tree to represent a source event type definition in the graphical user interface;

a module configured to display a second hierarchical tree to represent a destination event type definition in the graphical user interface;

a module configured to receive an indication of a selection from a pointing device;

a module configured to detect that a pointer in the graphical user interface that is manipulated by the pointing device is over a representation of a first node in the first hierarchical tree at the time that selection from the pointing device was initiated;

a module configured to detect that the pointer is guided from the representation of the first node in the first hierarchical tree to a representation of a second node in the second hierarchical tree, where the representation of the second node corresponds to a location of the pointer in the graphical user interface at the time that selection from the pointing device was terminated;

a module configure to draw a line in the graphical user interface from the representation for the first node in the first hierarchical tree to the representation of the second node in the second hierarchical tree; and

a module configured to automatically generate software code that maps data from the first node of the source event type definition to the second node of the destination event type definition.

38. The system as defined in Claim 37, wherein the module that is configured to draw the line from the representation for the first node in the first hierarchical tree to the representation of the second node in the second hierarchical tree further comprises:

a module configured to draw the line in the graphical user interface while the indication of the selection is present, where a first end of the line starts from about the first representation for the first node in the first hierarchy, and where a second end of the line ends about at the pointer such that the second end of the line moves with movement of the pointer; and

a module configured to fix the line at the second node of the second hierarchical tree when the indication of the selection is terminated.

39. The system as defined in Claim 37, wherein the graphical user interface further comprises:

a first pane configured to display a representation of the source event type definition; and a second pane configured to display a representation of the destination event type definition.

40. The system as defined in Claim 37, wherein the graphical user interface further comprises:

a first pane configured to display a representation of the source event type definition;

a second pane configured to display a representation of the destination event type definition; and

a third pane that is disposed between the first pane and the second pane, where the third pane displays a correspondence between a node of the source pane and a node of the destination pane via a line drawn between the first pane and the second pane.

41. The system as defined in Claim 37, wherein the graphical user interface further comprises a mapping pane disposed between a source event pane and a destination event pane, where a portion of a line in the mapping pane joins a portion of a line from the source event pane and a portion of a line from the destination event pane to illustrate a correspondence of data between a source event node and a destination event node.

42. A method of automatically generating software code for an application program interface (API) call in a graphical user interface, the method comprising:

receiving a selection of a source event type definition and a destination even type definition;

displaying trees that represent a hierarchical structure for the source event type definition and the destination even type definition;

detecting an indication of mapping from a source node to a destination node;

drawing a line in the graphical user interface to indicate the mapping; and

retrieving a structure for the source node and a structure for the destination node from the source event type definition and from the destination event type definition; and

generating a collaboration rule from the retrieved structures.

43. The method as defined in Claim 42, further comprising saving the generated collaboration rule.

44. The method as defined in Claim 42, further comprising:

providing a pane in the graphical user interface configured to show the generated software code; and

permitting the automatically generated source code to be edited within the pane for customizations.

45. A computer-readable medium having computer-executable instructions for automatically generating software code for an application program interface (API) call in a graphical user interface comprising:

receiving a selection of a source event type definition and a destination even type definition;

displaying trees that represent a hierarchical structure for the source event type definition and the destination even type definition;

detecting an indication of mapping from a source node to a destination node;

drawing a line in the graphical user interface to indicate the mapping; and

retrieving a structure for the source node and a structure for the destination node from the source event type definition and from the destination event type definition; and

generating a collaboration rule from the retrieved structures.

46. A method of modifying software code for an application program interface (API) call in a graphical user interface, the method comprising:

retrieving a collaboration rule;

retrieving a source event type definition and a destination even type definition corresponding to the collaboration rule;

displaying trees that represent a hierarchical structure for the source event type definition and the destination even type definition;

detecting a change in a mapping from a source node to a destination node;

updating lines in the graphical user interface to indicate the changes to the mapping;

retrieving a structure for the source node and a structure for the destination node from the source event type definition and from the destination event type definition; and

updating a collaboration rule from the retrieved structures.

47. The method as defined in Claim 46, wherein the collaboration rule is retrieved in response to a user selection.

48. The method as defined in Claim 46, further comprising saving the generated collaboration rule.

49. The method as defined in Claim 46, further comprising:

providing a pane in the graphical user interface configured to show the updated software code; and

permitting the updated source code to be edited within the pane for customizations.

50. A computer-readable medium having computer-executable instructions for modifying software code for an application program interface (API) call comprising:

retrieving a collaboration rule;

retrieving a source event type definition and a destination even type definition corresponding to the collaboration rule;

displaying trees that represent a hierarchical structure for the source event type definition and the destination even type definition;

detecting a change in a mapping from a source node to a destination node;

updating lines in the graphical user interface to indicate the changes to the mapping;

retrieving a structure for the source node and a structure for the destination node from the source event type definition and from the destination event type definition; and

updating a collaboration rule from the retrieved structures.